A stylized illustration of a tree with a thick black trunk and branches. The leaves are represented by numerous light green circles of varying sizes, some of which are falling from the branches. The background is a solid light green color.

# Habitat

Abdul Moiz  
Mohid Makhdoomi

Kareem Khaled  
Tyler Phillips

Jakub Pawlikowski

Group 3

# Project Objective, Scope and Motivation

---

- Main Goal:
  - To assist people who are moving or immigrating find a place to live with the highest quality of life
- Scope:
  - The implementation works for locations all across Canada
- Motivation:
  - Personal experience and interactions with peers



# Data Sets

---

- Datasets acquired from Statistics Canada
- Datasets:
  - Monthly Climate Summaries
  - Police Personnel and Selected Crime Statistics, Municipal Police Services
  - Population and Dwelling Count
  - Group, Income Taxes and Paid After-Tax Income

# Functional Requirements

---

- Add and Edit Locations to the Application
- Input Ranking values for Factors
- Sorting and Searching Algorithms to Access Datasets
- Combine Entered values for Factors with Data for Cities
- Rank the Quality of Life of each City
- Output the list of Cities ordered by Quality of Life



# Non-Functional Requirements

---

- Reliability
- Accuracy
- Robustness
- Performance
- Modularity
- Maintainability



# Main Design Specifications

---

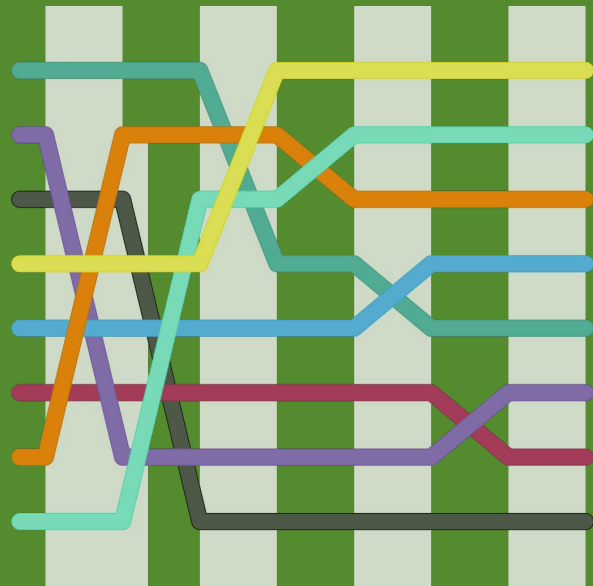
- Input:
  - The user inputs a list of Canadian cities they are interested in living in as well as how much they value each of the factors used to measure the quality of life (Crime rate, climate, population and average income)
- Output:
  - The inputted cities will be ordered based on their respective quality of life
- There is a lot of data to work with for all of the quality characteristics

This is where our algorithms come in!

# Our Algorithms

---

- Sorting:
  - Merge Sort :  $O(N(\log N))$
- Searching:
  - Binary Search  $O(\log N)$
- Graphing:
  - BST and Connected Components

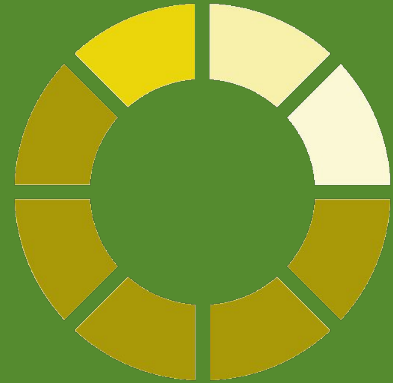


# Verification and Validation Methods

---

- Verification

- The correctness of our implementation was verified by unit testing each module separately
- The specification for the modules were drawn out before implementation to ensure correct implementation
- Concept -> Mathematical model -> Implementation -> Verified





# Verification and Validation Methods

---

- Validation
  - Are we building the right product?
  - Taking 3rd party opinions
  - Evaluating software from an unbiased perspective
  - Re establishing requirements and seeing if design needs to be changed
  - Comparing it to similar programs

# References

---

Population and Dwelling Count Highlight Tables, 2011 Census [Advertisement]. (n.d.). Retrieved March 31, 2017, from <http://www12.statcan.gc.ca/census-recensement/2011/dp-pd/hlt-fst/pd-pl/Table-Tableau.cfm?LANG=Eng&T=301&S=3&O=D>

Monthly Climate Summaries [Advertisement]. (n.d.). Retrieved March 31, 2017, from [http://climate.weather.gc.ca/prods\\_servs/cdn\\_climate\\_summary\\_e.html](http://climate.weather.gc.ca/prods_servs/cdn_climate_summary_e.html)

Police personnel and selected crime statistics, municipal police services [Advertisement]. (n.d.). Retrieved March 31, 2017, from <http://open.canada.ca/data/en/dataset/d666bc4a-d9b0-4b48-84de-dedb963716be>

Neighbourhood income and demographics, taxfilers and dependents with income, by sex, age group, income taxes paid and after-tax income [Advertisement]. (n.d.). Retrieved March 31, 2017, from <http://open.canada.ca/data/en/dataset/69ca450b-beeb-48a3-8685-b2d20e5432d4>